



# 廣東工業大學

## 课 程 设 计 报 告

课程名称 操作系统课程设计

学生学院 计算机学院

专业班级 人工智能 1 班

学 号 3121005358

学生姓名 欧炜标

指导教师 张伟文

2023 年 6 月 10 日



## 目录

1 课程目标.....	4
2 课程内容.....	4
3 课程设计.....	4
4 实现思路与分析.....	5
4.1 为宿主系统准备好搭建 LFS 目标系统的环境.....	5
4.2 以 lfs 用户构建临时工具链.....	6
4.3 在 chrooted 环境下构建 LFS 目标系统.....	7
4.4 配置 LFS 目标系统.....	7
4.5 编译并安装内核.....	7
4.6 设置 GRUB 使新系统可引导 .....	7
4.7 进入新系统.....	8
5 运行（系统运行截图） .....	8
6 心得体会.....	9

## 1 课程目标

学习掌握操作系统内核、基础库、工具链等的编译安装、文件系统构成及启动引导流程，培养学生动手实践能力;进入开源社区学习、开发，了解开源软件开发理念。

## 2 课程内容

本课程旨在让学生融入 openEuler 开源社区，以 openEuler 操作系统为开发环境，从 0 到 1 构建出一个属于自己的 Linux 操作系统。

## 3 课程设计

本创新实践课共分三个阶段，包括准备学习，开发实战活动以及总结作业提交;后期学生可以自主创新地完成进阶内容。

## 4 实现思路与分析

### 4.1 为宿主系统准备好搭建 LFS 目标系统的环境

#### 4.1.1 实验环境

实验设备	操作系统: Windows10 处理器: 11th Gen Intel(R) Core(TM) i7-11300H @ 3.10GHz 3.11 GHz 内存: 16GB
虚拟机软件	VirtualBox-6.1.44
虚拟机规格	逻辑处理器数:4 内存:8GB 磁盘大小:30GB
宿主机系统	openEuler 操作系统 20.09
LFS 版本	LFS-BOOK-7.7-systemd

#### 4.1.2 虚拟机的安装

安装两块硬盘：其中，sda 上安装了宿主操作系统 openEuler，sdb 则用作承载 LFS 目标系统的硬盘。硬盘的总线接口可以是 SCSI 或 SATA。在编译内核时，需要选择相关的驱动，以支持这两块硬盘。

同时，设置了两张网卡：一张使用默认的 NAT 模式，以便从虚拟机中访问外网；另一张使用 Host-only 模式，以便从宿主机访问该虚拟机。这样可以实现虚拟机与宿主机之间的网络连接。

#### 4.1.3 分区及格式化

对于 sdb，进行了分区操作，创建了一个主分区 sdb1，用于承载 LFS 系统。现在有了一个空的分区，可以创建文件系统。LFS 可以使用任何 Linux 内核能识别的文件系统类型，但最常用的是 ext3 和 ext4。在这种情况下，选择了 ext4 作为 LFS 根文件系统的类型。完成文件系统的创建后，需要对该磁盘进行格式

化, 这样 LFS 目标系统就可以在此处进行编译和安装。

文件系统创建完毕后, 需要将分区挂载到选定的挂载点上以进行访问。假设挂载点为 `/mnt/fs`。创建了一个 LFS 变量, 并将其指向宿主系统的 `/mnt/lfs/` 目录。然后将 `sdb1` 分区挂载到该目录上。

## 4.2 以 lfs 用户构建临时工具链

在 LFS 官网下载页面找到“Packages for LFS”一节, 从中下载了 `lfs-packages-7.7-systemd.tar` 包到本地。为了在整个构建过程中能够有效地访问这些下载好的软件包和补丁, 需要将它们保存在一个特定的位置, 并准备一个工作目录用于解压源代码并进行编译。根据实验指导书的要求, 我们选择将这些软件包和补丁保存在名为 `LFS/sources` 的目录中, 并将其作为工作目录。通过将它们放置在这个目录中, 所有需要的元素都可以在 LFS 分区中找到, 因此在构建过程的所有阶段都可以访问到它们。

在 SLFS 中, 可以创建一个新文件夹来安装临时工具。所有在第五章中编译的软件都将被安装到 `SLFS/tools` 文件夹中, 以确保与第六章中编译的软件相互分离。这些软件是临时工具, 不会成为最终 LFS 系统的一部分。将它们保存在单独的文件夹中可以方便地在使用完成后进行丢弃。

增加一个非特权用户 LFS 用于降低风险, 并为该用户创建合适的构建环境。当以 `root` 用户登录时, 犯一个小错误可能会破坏或摧毁整个系统。因此为了建立一个干净的工作环境, 创建一个名为 `lfs` 的新用户作为新组(名字也是 `lfs`)的成员, 并在安装过程中使用这个用户。

当以“LFS”用户身份登录时, 初始的 `shell` 通常是一个可登录的 `shell`。它首先会读取宿主机的 `/etc/profile` 文件, 其中可能包含一些设置和环境变量, 然后读取 `“.bash_profile”` 文件。在 `“.bash_profile”` 文件中, 通过 `“exec env -i.../bin/bash”` 命令, 使用一个完全空的环境(除了 `HOME`、`TERM` 和 `PS1` 变量)来替代当前运行的 `shell`。这样可以确保没有不必要的或潜在风险的环境变量从宿主机系统泄露到构建环境中。这种技巧用于创建一个干净的环境。

新的 `shell` 实例是一个非登录 `shell`, 它不会读取 `/etc/profile` 或者 `“.bash_profile”` 文件, 而是读取 `“.bashrc”` 文件。因此, 我们需要创建一个自定义的 `“.bashrc”` 文件来

配置我们希望的环境。

## 4.3 在 chrooted 环境下构建 LFS 目标系统

为了确保目标 LFS 系统的洁净和无故障生成，我们将使用 `chroot` 命令将后续的命令包含在一个特定的环境中。`chroot` 命令可以被视为对宿主系统的简单虚拟化。在 `chroot` 环境中进行操作相当于操作 LFS 目标系统的位置。通过在 `chroot` 环境中构建 LFS 目标系统的系统软件，可以避免在构建过程中对宿主系统造成损坏。

为了确保 `chroot` 环境中能够访问到所有源文件和补丁，需要将它们放置在一个可访问的目录，比如 `/mnt/lfs/sources/`。首先，进入源文件目录，然后使用 `tar` 程序解压要编译的软件包。请确保在解压软件包时使用 `lfs` 用户，并进入解压后创建的目录。按照指南中的说明编译软件包。完成后，返回到源文件目录，并在没有特殊说明的情况下删除解压出来的目录和所有编译过程中生成的 `-build` 目录。最后，编译和安装基本的系统软件后，进行清理以节省空间。

## 4.4 配置 LFS 目标系统

为 LFS 系统进行网络、主机名、`/etc/hosts` 文件的设置。

## 4.5 编译并安装内核

在编译内核时，需要按照以下步骤进行配置、编译和安装，所有命令都要在 `chroot` 环境下运行。在 `"make menuconfig"` 这一步中，除了 LFS-BOOK 中提到的配置选项外，还需要选择与虚拟机提供的硬件相匹配的驱动，特别是硬盘的驱动。否则，启动新系统时可能无法挂载 `sdb`。根据我使用的虚拟机软件选择相应的驱动。由于我使用的是 `VirtualBox` 虚拟机软件，默认选项是适用的。

## 4.6 设置 GRUB 使新系统可引导

在 LFS 系统(即目前的 `chrooted` 环境) 生成 `grub.cfg` 文件，将 LFS 系统的 `grub.cfg` 中 `menuentry` 模块复制到宿主机 `/boot/grub2/grub.cfg` 中的相应位置，

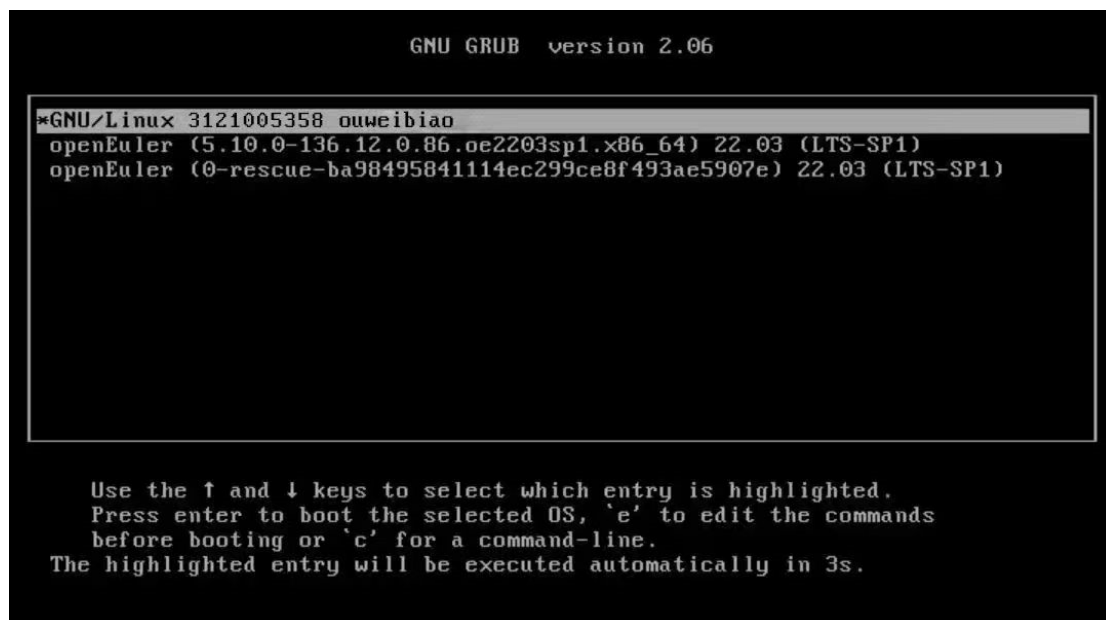
更改这条菜单项的标题为“GNU/Linux 3121005362 ouweibiao”。

## 4.7 进入新系统

重启系统，在引导界面就可以见到 LFS 系统，选中并确认，输入用户名和密码即可进入新系统。

## 5 运行（系统运行截图）

【系统启动界面】



【系统运行界面】



```
RX
[ 4.084677] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[ 4.201848] hrtimer: interrupt took 27323654 ns
root
Password:
No mail.
-bash-4.3# uname -m
x86_64
-bash-4.3# uname -r
3.19.0
-bash-4.3# cat /etc/os-release
NAME="Linux From Scratch"
VERSION="7.7-systemd"
ID=lfs
PRETTY_NAME="Linux From Scratch 7.7-systemd"
-bash-4.3# cat /etc/lfs-release
7.7-systemd
-bash-4.3# cat /etc/lsb-release
DISTRIB_ID="Linux From Scratch"
DISTRIB_RELEASE="7.7-systemd"
DISTRIB_CODENAME="3121005358- ouweibiao "
DISTRIB_DESCRIPTION="Linux From Scratch"
-bash-4.3# hostname
3121005358- ouweibiao
-bash-4.3#
```

## 6 心得体会

通过构建 LFS 系统，我获得了以下收获和经验：

**深入理解 Linux 内部工作原理：**构建 LFS 系统使我更深入地了解了 Linux 操作系统的内部工作原理。我学会了如何从头开始构建一个完整的 Linux 系统，并深入研究了操作系统的各个组成部分以及它们之间的交互和协作。

**掌握核心概念和机制：**通过构建 LFS 系统，我深入学习了操作系统的核心概念和机制，包括进程管理、内存管理、文件系统和设备驱动程序等。我了解了这些概念和机制是如何相互作用和协同工作的。

**实践操作系统原理：**构建 LFS 系统是将操作系统原理的理论知识应用于实际项目的一种方式。通过亲自参与系统的构建和配置，我深入理解了各个组件的功能和相互关系，并学会了如何应用这些知识来构建和优化一个完整的系统。

**增强问题解决能力：**在构建 LFS 系统的过程中，我面临了各种问题和挑战。通过解决这些问题，我提升了自己的问题解决能力。我学会了分析问题、查找解决方案，并灵活地应用这些解决方案来解决实际问题。

**学习系统调优和性能优化：**构建 LFS 系统使我学会了系统调优和性能优化的技巧。我学会了根据系统需求选择合适的软件包，并通过配置系统参数和优化编

译选项来提高系统的性能和资源利用率。

通过构建 LFS 系统，我不仅加深了对操作系统的理解，还培养了解决问题和优化系统的能力。这个实践经验为我在操作系统领域的进一步学习和探索打下了坚实的基础，并激发了我对操作系统的兴趣和热情。